

## WEST Search History

[Hide Items](#)[Restore](#)[Clear](#)[Cancel](#)

DATE: Sunday, August 22, 2004

Hide?	<u>Set</u> <u>Name</u>	<u>Query</u>	<u>Hit</u> <u>Count</u>
		<i>DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ</i>	
<input type="checkbox"/>	L4	((Java adj3 machine) or JVM) near8 (object adj3 code) near8 runtime	2
<input type="checkbox"/>	L3	((Java adj3 machine) or JVM) near8 (object adj3 code) near8 (runtime adj3 operating)	0
<input type="checkbox"/>	L2	L1 and (dependent adj4 function)	3
<input type="checkbox"/>	L1	(operating adj3 system) near8 (independent near4 interface)	181

END OF SEARCH HISTORY

BEST AVAILABLE COPY

[Previous Doc](#)   [Next Doc](#)   [Go to Doc#](#)  
[First Hit](#)

☐ **Generate Collection**

L4: Entry 1 of 2

File: PGPB

Jan 22, 2004

DOCUMENT-IDENTIFIER: US 20040015852 A1

TITLE: System and method for transforming object code

Summary of Invention Paragraph:

[0005] Once a program is written in Java source code, the Java compiler generates a compact, architecture-neutral object code (commonly referred to as known as Java bytecode) which may be executed by a runtime interpreter residing on the client computer. This runtime interpreter is commonly referred to as a Java "virtual machine." The Java virtual machine interprets the object code so that the instructions may be executed by the client's native microprocessor. Virtual machines are included in commonly available Internet browser applications such as Netscape Navigator.TM. or Microsoft Internet Explorer..TM.

[Previous Doc](#)   [Next Doc](#)   [Go to Doc#](#)

BEST AVAILABLE COPY

[Previous Doc](#)   [Next Doc](#)   [Go to Doc#](#)  
[First Hit](#)   [Fwd Refs](#)

☐ [Generate Collection](#)

L2: Entry 2 of 3

File: USPT

Jun 9, 1998

DOCUMENT-IDENTIFIER: US 5765039 A

TITLE: Method for providing object database independence in a program written using the C++ programming language

Detailed Description Text (74):

This is accomplished in a somewhat database dependent means in the implementation of the member functions in the surrogate class. If the database contains an operation router, and that router properly supports polymorphism, then the router is simply called to do the work. If the router does not handle polymorphism, or there is no router, an alternate tack must be taken.

CLAIMS:

1. In a computer system having a user interface, a CPU, a memory, at least one disk drive, and an object oriented database stored in one or more disk drives, a program operating in said computer system for deriving an application programming interface independent from objects stored in said database, said program executing for each database type, TYPE, a method comprising the steps of:

a. opening a file on said at least one disk drive and storing therein results of the following steps;

b. declaring a surrogate class, sTYPE, for a database type, TYPE;

c. for each property, PROP, of TYPE, declaring accessors and mutators;

d. declaring construct and destruct member functions;

e. for each operation, OP, of TYPE, declaring member functions; and,

f. declaring special functions.

13. In a computer system having a user interface, a CPU, a memory, at least one disk drive, and an object oriented database stored in one or more disk drives, a program operating in said computer system for deriving an application programming interface independent from objects stored in said database, said program executing for each database type, TYPE, a method comprising the steps of:

a. opening a file on said at least one disk drive and storing therein results of the following steps;

b. declaring a surrogate class, sTYPE, for a database type, TYPE, if TYPE is a base type, sTYPE inheriting publicly for said surrogate class; if TYPE is not a base type, sTYPE inheriting from surrogate class that corresponds to database supertype; declaring a public default constructor that has no parameters; declaring a public copy constructor; and, declaring a public virtual destructor;

c. for each property, PROP, of TYPE, declaring accessors and mutators;

**BEST AVAILABLE COPY**

- d. declaring construct and destruct member functions;
- e. for each operation, OP, of TYPE, declaring member functions; and,
- f. declaring special functions.

[Previous Doc](#)[Next Doc](#)[Go to Doc#](#)

BEST AVAILABLE COPY